

---

# CAGE CHALLENGE 3 SUBMISSION

---

CYBER AUTONOMY GYM FOR EXPERIMENTATION (CAGE)

**Noor El Alfi**  
School of Computing  
Queen's University  
Kingston, Ontario  
noor.elalfi@queensu.ca

**Dr. Steven Ding**  
School of Computing  
Queen's University  
Kingston, Ontario  
steven.ding@queensu.ca

February 17, 2023

## 1 Overview

Team Name:	L1NNA
Agent Name:	PPO Agent
Challenge Problem:	CAGE Challenge 3
Primary Model(s):	PPO
Source:	<a href="https://github.com/NoorElAlfi/CC3_sub">https://github.com/NoorElAlfi/CC3_sub</a>

## 2 Approach

PPO was used to train the blue team agents, the reason we decided to use PPO is because according to publications within the field PPO was found to have competitive and/or superior in final returns and sample efficiency compared to competing models. PPO also requires minimal hyper-parameter tuning and no domain specific algorithmic modifications or architectures. Our readings of studies using PPO for MARL also revealed that simple PPO-based methods are a strong baseline in multi-agent reinforcement learning.

### 2.1 Model

DDQN, RDQN, PPO RNN, and PPO were all implemented and tested, and it was found that on average the PPO model produced the best results. As a result PPO was the model we decided on.

Hyperparameters:

Learning rate	0.002
Epochs	5000
gamma	0.99
betas	0.9, 0.990

Table 1: Model Hyperparameters

## 2.2 Training

Our PPO model was trained using 5000 epochs where each epoch consisted of 1000 episodes. The agents that lost connection with the drone support team were penalized -10 for their failure.

## 2.3 Evaluation

Our trained model was run 10 times in the environment consisting of 100 epochs and 1000 episodes/epoch. The results are as follows:

Table 2: Runs

Run	Average reward	Total reward
Run 1	-4945.16	-2330.0
Run 2	-5627.98	-3118.0
Run 3	-5663.11	-3797.0
Run 4	-5697.66	-3467.0
Run 5	-5679.04	-6052.0
Run 6	-5363.0	-1114.0
Run 7	-5346.76	-6145.0
Run 8	-5207.38	-2892.0
Run 9	-5295.0	-5996.0
Run 10	-5750.67	-3746.0

## 3 Instructions

### 3.1 Requirements

Using CybORG default requirements along side pytorch.

```
pip install -r requirements.txt
```

### 3.2 Usage

Training:

```
./train.py
```

Evaluation:

```
./evaluation.py
```

## References

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu The Surprising Effectiveness of PPO in Cooperative, Multi-Agent Games <https://doi.org/10.48550/arXiv.2103.01955>, 2021.